



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/494,218	01/28/2000	Jason M Brewer	TI-28385	3161
23494	7590	08/24/2007		
TEXAS INSTRUMENTS INCORPORATED			EXAMINER	
P O BOX 655474, M/S 3999			BLAIR, DOUGLAS B	
DALLAS, TX 75265				
			ART UNIT	PAPER NUMBER
			2142	
			NOTIFICATION DATE	DELIVERY MODE
			08/24/2007	ELECTRONIC

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

uspto@ti.com  
uspto@dlemail.itg.ti.com



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents  
United States Patent and Trademark Office  
P.O. Box 1450  
Alexandria, VA 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

MAILED

AUG 24 2007

Technology Center 2100

**BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES**

Application Number: 09/494,218

Filing Date: January 28, 2000

Appellant(s): BREWER, JASON M

Carlton H. Hoel (Reg. No. 29,934)  
For Appellant

**EXAMINER'S ANSWER**

This is in response to the appeal brief filed 7/16/2007 appealing from the Office action mailed 12/28/2006.

**(1) Real Party in Interest**

A statement identifying by name the real party in interest is contained in the brief.

**(2) Related Appeals and Interferences**

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

**(3) Status of Claims**

The statement of the status of claims contained in the brief is correct.

**(4) Status of Amendments After Final**

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

**(5) Summary of Claimed Subject Matter**

The summary of claimed subject matter contained in the brief is correct.

**(6) Grounds of Rejection to be Reviewed on Appeal**

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

**(7) Claims Appendix**

The copy of the appealed claims contained in the Appendix to the brief is correct.

**(8) Evidence Relied Upon**

6,295,638	BROWN et al.	9-2001
6,389,589	MISHRA	5-2002
6,263,360	ARNOLD et al.	7-2001

**(9) Grounds of Rejection**

The following ground(s) of rejection are applicable to the appealed claims:

Claim 5 is rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent Number 6,295,638 to Brown et al..

As to claim 5, Brown teaches a system for loading Java class file to a client device (col. 2, line 50-67, and col. 5, lines 38-54) comprising: a gateway (col. 7, line 26, the compiler process is the gateway) coupled to said server and responsive to a Java class file for creating a c-code representation of said class file (col. 7, lines 25-44, The front end compiler creates a c-code representation.); said gateway creating a binary representation of said c-code representation (col. 7, lines 25-44, The backend compiler creates optimized content.); a network coupled between said gateway and said client device for sending the binary representation to said client device (col. 5, lines 38-54 and col. 9, lines 23-37); a loader for loading said binary representation at said client device (col. 8, lines 20-61); and, means for copying said binary representation into the internal class structure in an interpreter of said client device (col. 8, lines 62-67 and col. 9, lines 1-15).

Claims 1-3 and 6-7 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent Number 6,295,638 to Brown et al. in view of U.S. Patent Number 6,389,589 to Mishra et al..

As to claim 1, Brown teaches a method for loading class files from a server to a client (col. 2, lines 50-67) comprising: loading an application class onto a gateway server that preloads and preresolves said class (col. 7, lines 25-44, The front end compiler preloads and preresolves the classes.); creating a binary representation of new portions of the preloaded and preresolved

class at said gateway (col. 7, lines 25-44, The backend compiler creates optimized content.); however Brown does not explicitly teach sending only the new portion to the client.

Mishra teaches a method of sending only the new portions of application classes to the client (col. 17, lines 36-62, Only upgrade components are sent to the client.).

It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Brown regarding a method for loading class files with the teachings of Mishra regarding a method for sending only new portions of classes to a client because sending a smaller amount of data conserves bandwidth.

Brown teaches the invention substantially as claimed (e.g. exemplary claim 7) including a method for loading Java class files to an embedded client device from a server (col. 2, lines 50-67) comprising the steps of: gateway retrieving a Java class file, gateway preloading and preresolving the Java class file to produce a representation of the Java class file (col. 7, lines 25-44, The front end compiler preloads and preresolves the classes.); creating at the gateway a binary representation of only said new portion of the preloaded and preresolved representation of the Java class file (col. 7, lines 25-44, The backend compiler creates optimized content.); sending said binary representation into said embedded client device (col. 5, lines 38-54); and, copying said binary representation into the internal class structures in the interpreter of a Java virtual Machine of the embedded client (col. 8, lines 62-67 and col. 9, lines 1-15); however Brown does not explicitly teach a method of determining at the gateway a new portion of the representation and forwarding only the new .

Mishra teaches a method of determining at the gateway a new portion of the representation (col. 17, lines 36-62, The server determines which client components to upgrade.).

It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Brown regarding a method for loading Java classes with the teachings of Mishra regarding the method of making a determination of which class to load because sending a smaller amount of data conserves bandwidth.

As to claim 2, it comprises the same steps as claim 7 with a broader preamble therefore it is rejected on the same basis as claim 7.

As to claim 3, Brown-Mishra teaches the method of claim 2 including determining new portions of a class representation. Brown teaches creating a c-code representation of the Java class file (col. 7, lines 25-44, The front end compiler creates a c-code representation.), and creating a binary representation of said c-code representation; however Brown does not explicitly teach a method for determining new portions or creating binaries of only new portions.

Mishra teaches a method of determining new portions of code and creating binaries of the new portions (col. 17, lines 36-62).

It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Brown regarding a method for loading Java classes with the teachings of Mishra regarding the method of making a determination of which class to load because sending a smaller amount of data conserves bandwidth.

As to claim 6, Brown teaches the system of claim 5; however Brown does not explicitly teach a system for determining new portions of the c-code representation or sending only new portions of the c-code representations.

Mishra teaches a system including a means for determining new portions of a c-code representation, and a means for creating binary representations of only new portions of the c-

code representations, and a means for sending only the new portions to a client (col. 17, lines 36-62).

It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Brown regarding a method for loading Java classes with the teachings of Mishra regarding the method of making a determination of which class to load because sending a smaller amount of data conserves bandwidth.

Claims 4 and 9 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent Number 6,295,638 to Brown et al. in view of U.S. Patent Number 6,389,589 to Mishra et al. as applied to claim 2 above, and further in view of U.S. Patent Number 6,263,360 to Arnold et al..

As to claim 4, the teachings of Brown-Mishra combine to make claim 2 obvious; however Brown and Mishra do not explicitly teach sending the classes over a wireless network.

Arnold teaches a method of sending Java classes over a wireless network (col. 26, lines 25-67).

It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Brown-Mishra regarding a method for loading Java classes with the teachings of Arnold regarding a method for sending Java classes over a wireless network because Java is a common tool for developing wireless applications due to its platform independence (col. 24, lines 51-63 of Arnold).

As to claim 9, the limitations of claim 9 are rejected for the same reasons as claim 7.

Claim 8 is rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent Number 6,295,638 to Brown et al. in view of U.S. Patent Number 6,263,360 to Arnold et al..

As to claim 8, it has the same limitations as claim 5 with the additional limitation of sending classes over a wireless network. Brown does not teach sending classes over a wireless network.

Arnold teaches a method of sending Java classes over a wireless network (col. 26, lines 25-67).

It would have been obvious to one of ordinary skill in the Computer Networking art at the time of the invention to combine the teachings of Brown regarding a system for loading Java classes with the teachings of Arnold regarding a system for sending Java classes over a wireless network because Java is a common tool for developing wireless applications due to its platform independence (col. 24, lines 51-63 of Arnold).

#### **(10) Response to Argument**

The appellant's only argument is that there is not a gateway in Brown. The appellant states that the compilers in Brown do not read on the gateway claimed by the applicant. The appellant argues that the term "gateway" is standard telecommunications terminology, and a gateway at minimum connects two networks which have different protocols. The appellant then states that, "Application Fig. 3 illustrates this with gateway 37 on wired network 41 with server 31 and connecting to wireless network 43 with client 45."

The Examiner does not agree with the appellant's narrow view of a gateway. The appellant does not disclose a gateway that connects two networks having two different protocols. The appellant's specification does not mention any protocols or any translation between protocols. The appellant's own specification states that, "The client 35 loads the application

through a **gateway 37 at or wired to the server** at the server location" (Appellant's specification, page 6, lines 12-14). If the gateway can be at the server in Figure 3, then the "wired network 41" is not even necessary for the appellant's claimed gateway to function. In view of the appellant's disclosure it is more appropriate to interpret the claimed gateway according to the more generic American Heritage Dictionary definition of a gateway which is "something that serves as an entrance or a means of access".

Brown teaches a server featuring a compiler process that functions as the claimed gateway as shown in the mapping in the rejection of claim 5 based on Brown (Shown in the previous section). The server in Brown sends compiled Java files as FCCF code to the client. The compiler process is considered a "gateway" because it serves as the entrance for Java class files that are to be converted into FCCF code format and sent to the client. This interpretation is consistent with the appellant's broad disclosure of a gateway which in no way limited to what the appellant is now arguing a gateway is.

#### **(11) Related Proceeding(s) Appendix**

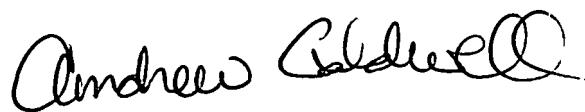
No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Douglas Blair  
DBB

Conferees:



ANDREW CALDWELL  
EXAMINER



JASON CARDONE  
SUPERVISORY PATENT EXAMINER